

10/511188

CT04 Rec'd PCT/PTO 12 OCT 2004

CLAIM AMENDMENTS

1 1. (Currently amended) ~~Method for realising~~ A method
2 managing a management activity of at least one managed object (Bi,
3 ..., BN) by at least one manager object through a communication
4 network (R), characterized in that it comprises the following
5 steps:

6 - providing at least one intermediate object (AG)
7 configured to manage said at least one managed object (B1, ..., BN)
8 according to a data set (1102), said management activity being
9 transformed into a set of results (1112),

10 - providing said data set (1100) from said at least one
11 manager object (A) to said intermediate object (AG),

12 - managing said at least one managed object (B1, ...,
13 Bn) through said at least one intermediate object (AG), to generate
14 said set of results, and

15 - transferring {1108} said set of results from said at
16 least one intermediate object (AG) to said at least one manager
17 object (A).

1 2. (Currently amended) Method The method according to
2 claim 1, ~~characterised in that it~~ which comprises the step of
3 establishing the communication between said at least one manager

object (A) and said at least one intermediate object via UDP protocol.

3. (Currently amended) Method The method according to claim ~~1 or claim 2~~, characterised in that it which comprises the following steps:

- managing at least one further managed object (Bk, 25 BN) directly through said at least one manager object (M, and
- managing said at least one managed object (Bi, B2, B3) by said at least one manager object (A) via said intermediate object (AG).

4. (Currently amended) Method The method according to claim 3, ~~characterised in that it~~ comprises the management of said at least one further managed object (Bk, ..., Bn) and said at least one managed object (B1, B2, B3) through a single communication network (R).

5. (Currently amended) Method The method according to claim 3, ~~characterised in that it~~ which comprises the following steps:

4 - providing a first communication network (RP) for
5 managing said at least one further managed object (B1) directly
6 through said at least one manager object (A) and transferring said
7 data set (1100) and said results set (1118) between said at least
8 one manager object (A) and said at least one further managed object
9 (Bi), and

10 - providing a second communication network (RA) for
11 managing said at least one managed object (B2, B3) through said
12 intermediate object (AG).

1 6. (Currently amended) Method The method according to
2 ~~any of the previous claims, characterised in that it~~ claim 3 which
3 comprises the steps of providing a plurality of said intermediate
4 objects (AG1, AG2) and managing at least one managed object (B3)
5 through several intermediate objects (AG1, AG2) of said plurality.

1 7. (Currently amended) Method The method according to
2 ~~any of the previous claims, characterised in that~~ claim 3 wherein
3 said intermediate object (AG) is provided with respective reception
4 modules (AU) and transmission modules (ATX) configured so that said
5 at least one manager object (A) sees said intermediate object (AG)
6 essentially as one of said managed objects (Bi, ..., Bn).

1 8. (Currently amended) Method The method according to any
2 ~~of the previous claims, characterised in that claim 3 wherein~~ said
3 at least one intermediate object (AG) comprises at least one
4 respective management module (MM) configured so that said at least
5 one managed object (B1, .. , BN), which is managed by said at least
6 one intermediate object (AG), sees said at least one intermediate
7 object (AG) essentially as said at least one manager object (A).

1 9. (Currently amended) Method The method according to any
2 ~~of the previous claims, characterised in that claim 3 wherein~~ said
3 at least one intermediate object (AG) is provided with one of the
4 following queues:

5 - an input queue (i) for collecting input messages with
6 respect to said at least one intermediate object (AG),

7 - an output queue (U) for collecting output messages from
8 said at least one intermediate object (AG), and

9 - a working queue (L) for collecting messages inherent
10 to said management activity performed by said at least one
11 intermediate object (AG) on said at least one managed object
12 (B1,Bn) .

1 10. (Currently amended) Method The method according to
2 claim 9 ~~, characterised in that it which~~ comprises the step of
3 providing, in said at least one intermediate object (AG), a
4 dedicated module (DC) for analyzing the input messages received by
5 said input queue (I).

1 11. (Currently amended) Method The method according to
2 claim 10 which ~~9 or claim 10, characterised in that it~~ comprises
3 the following steps:

4 - providing, in said at least one intermediate object
5 (AG), an activity co-ordinating module (CA) for
6 implementing at least one of the following functions:

7 - instantiating at least one concurrent process,
8 - updating activity status of the requests in said
9 working queue L, and
10 - creating statistic check messages to be sent to said
11 at least one manager object (A) through said output queue (U).

1 12. (Currently amended) Method The method according to
2 ~~any of the previous claims, characterised in that it~~ claim 9 which
3 comprises the step of providing a plurality of protocol management
4 modules (MP1, MP2, MP3) configured to establish communication to

5 said at least one managed object {B1, BN) through respective
6 different protocols in said at least one intermediate object (AG).

1 13. (Currently amended) Method The method according to
2 ~~any of the previous claims, characterised in that it claim 9 which~~
3 comprises the step of establishing the communication between said
4 at least one manager object (A) and said at least one intermediate
5 object (AG) by subjecting at least one part of the respective
6 messages to a compression operation {302; 104, 204).

1 14. (Currently amended) Method The method according to
2 claim 13, ~~characterised in that wherein~~ said compression operation
3 is based on the acknowledgment of a sequence which appears
4 periodically in the message.

1 15. (Currently amended) Method The method according to
2 claim 14, ~~characterised in that wherein~~ said compression operation
3 implements a gzip type method, such as zLib.

1 16. (Currently amended) Method The method according to
2 claim 2 ~~and any of the claims from 13 to 15, characterised in that~~
3 ~~it which~~ comprises the step of indicating that compression of the
4 message transferred by UDP is done.

1 17. (Currently amended) Method The method according to
2 ~~claim 16, characterised in that~~ wherein a bit field in the UDP
3 header is used to indicate that the compression operation (302) is
4 done.

1 18. (Currently amended) Method The method according to
2 ~~claim 17, characterised in that~~ wherein bits comprised in the range
3 from bit 62 to bit 69 in the UDP header are used in indicate that
4 the compression operation (302) is done.

1 19. (Currently amended) Method The method according to
2 ~~claim 18, characterised in that~~ which comprises the step of setting
3 at least one of the bits from 62 to 69 of the UDP message header to
4 1.

1 20. (Currently amended) Method The method according to
2 ~~any of the claims from 13 to 19, characterised in that~~ claim 13
3 wherein the communication between said at least one manager object
4 (A) and said at least one intermediate object (AG) is implemented
5 by means of SNMP messages, and comprises the following steps during
6 the compression step:

7 - reading (100) the entire SNMP message,

- 8 - encoding (102) the read message in hexadecimal 30
9 format, and
10 - subjecting the message encoded in hexadecimal format
11 to compression (104).

1 21. (Currently amended) Method The method according to
2 ~~any of the claims from 13 to 19, characterised in that claim 13~~
3 wherein communication between said at least one manager object (A)
4 and said at least one intermediate object (AG) is implemented by
5 means of SNMP messages, comprises the following steps during the
6 reception step:

- 7 - subjecting the received message to decompression (204)
8 complementary to said compression operation, to obtain a message
9 subjected to decoding in hexadecimal format,
10 - decoding (202) the message from the hexadecimal 10
11 format, and
12 - reconstructing (200) the entire SNMP message from said
13 decoded message.

1 22. (Currently amended) Method The method according to
2 claim 21 which ~~20 or claim 21, characterised in that it comprises a~~
3 nesting operation in a standard SNMP message for the transmission
4 of the message subjected to said compression operation (104).

1 23. (Currently amended) Method The method according to
2 claim 22 ~~[[,]] characterised in that it~~ which comprises the
3 following steps during transmission:

4 - reading (108) the message subjected to said
5 compression operation (104) in bytes and transposing (110) it into
6 a corresponding ASCII character message,

7 - generating (112) a variable binding set comprising a
8 first OID indicating the original file size and subsequent
9 OID/value pairs which carry portions of said message subjected to
10 said compression operation (104) transposed into ASCII characters,

11 - reconstructing SNMP message header data,

12 - encoding (114) the resulting SNMP message in
13 hexadecimal format to generate the UDP payload, and transferring
14 (116) the UDP payload generated in this way.

1 24. (Currently amended) Method The method according to
2 ~~claim 22 or claim 23, characterised in that it~~ 23 which comprises
3 the following steps during reception:

4 - receiving the message subjected to said compression
5 operation as an UDP payload (216),

6 - subjecting the payload received in this way to a
7 hexadecimal decoding operation (214),

8 - acknowledging and assembling (212) the variable
9 binding of the message subjected to hexadecimal decoding,
10 - subjecting the message subjected to said acknowledging
11 and assembling operation (212) to binary ASCII decoding (210), and
12 - subjecting the decoded message in binary form to said
13 decompression operation (204).

1 25. (Currently amended) Method The method according to
2 ~~claim 20 or claim 21, characterised in that it~~ 21 which comprises
3 the step of integrating the message subjected to said compression
4 operation (104) through UDP nesting for the transmission of the
5 message subjected to said compression operation (104).

1 26. (Currently amended) Method The method according to
2 ~~claim 25, characterised in that it~~ which comprises the following
3 steps during transmission:

4 - configuring said message subjected to said compression
5 operation (104) as a Protocol Data Unit (PDU)
6 payload, and
7 - transferring the payload created in this way to a
8 given receiver port.

1 27. (Currently amended) Method The method according to
2 ~~claim 25 or claim 26, characterised in that it~~ 26 which comprises
3 the following steps during reception:

- 4 - receiving said message as a payload of a PDU UDP
5 received at a receiver port, and
6 - extracting said payload from said PDU.

1 28. (Currently amended) Method The method according to
2 ~~claim 26 or claim 27, characterised in that it~~ 27 which comprises
3 the step of transmitting a synchronisation message (1106) of the
4 SNMP type indicating said transmission port and/or said reception
5 port between said at least one manager object (A) and said at least
6 one intermediate object (AG).

1 29. (Currently amended) System A system for managing
2 communication networks comprising at least one manager object (A)
3 and at least one managed object (B1, Bn), ~~characterised in that it~~
4 which comprises at least one intermediate object (AG) implementing
5 the method according to ~~any of the claims from 1 to 28~~ claim 3.

1 30. (Currently amended) ~~Software~~ A software module [[s]]
2 which can be directly loaded into the internal memory of at least a
3 computer and comprising portions of software code to implement the
4 method according to ~~any of the claims from 1 to 28~~ claim 3 when the
5 software modules are run by at least one computer.